5E-03

トークン連動型分散ファイルシステムの提案

大橋　盛徳† 渡邊　大喜† 石田　達郎† 藤村　滋† 中平　篤†　岸上　順一‡
日本電信電話株式会社 NTT サービスエボリューション研究所†　室蘭工業大学‡

1. はじめに

ブロックチェーン[1][2]によって、トラストレスにデジタルコンテンツなどのデジタル資産をトークン化し、不特定多数のユーザ間で流通可能とする環境が整いつつある。しかしながら、デジタル資産本体のファイルやデジタル資産に付随するファイルをブロックチェーンに登録すると台帳の肥大化という問題を引き起こしてしまう。ファイルをブロックチェーンではなく、既存のクラウドストレージに保存する回避方法は、Eberhardt ら[3]が指摘するように特定サービサーを信頼し依存することとなり、サービス継続上の単一障害点を生む。単一障害点のないデジタル資産流通システムを構築するためには、トークンだけでなく、そのトークンに紐づくファイルも特定サービサーに縛られない分散管理ができる必要がある。

ブロックチェーン外に保存する情報を分散的に管理する方法として InterPlanetary File System(IPFS)[5]などのトラストレスな分散ファイルシステムを活用する方法がある。従来の活用方法は、特定多数の間での流通を想定しており、分散ファイルシステムに保存したファイルの ID をブロックチェーン上のコントラクトに記録し、ブロックチェーンを通じてファイルの情報を共有している。

本研究では、従来の手法を拡張し、不特定多数のユーザ間の流通を想定した方式を提案する。

2. 関連研究と着眼点

ファイルをブロックチェーンではなく、分散ファイルシステムに保存し、ブロックチェーンと連携して利用する方式は、ファイルの流動性から 3 種類に分類される。1 つ目が、分散ファイルシステムにファイルを登録し、そのファイル ID をブロックチェーンに記録するという最もシンプルな方法である。この場合、ファイルは自由に流通し、流動性は高い。2 つ目は分散型クラウドストレージである。既存の分散型クラウドストレージには Storj[2], Sia[3], Filecoin[4]などがある。分散型クラウドストレージは、複数のストレージ提供者とファイル保存の契約を行い、保存してもらう方法である。この場合、ファイルの移動は制限されるため流動性は低くなる。3 つ目が、上記 2 つの中間の流動性となるもので、ファイルの流通をある条件下で認めるものである。Steichen ら[4]の方式はブロックチェーン上の特定コントラクトにファイルの共有条件を記載し、その条件に従うように分散ファイルシステム上のファイルの共有を制御している。

単一障害点のないデジタル資産流通システムには、流通を制御できる 3 つ目の中間の流動性を持つ方式が必要である。既存の方式[4]は、特定のコントラクトに制御情報を集約する設計となっており、管理の分散性よりも管理の容易性を重視した設計となっている。既存の方式をファイル毎やファイル群毎に別々のコントラクトで管理できるように拡張することにより、我々が目指す、単一障害点のない不特定多数のユーザのためのデジタル資産流通システムが可能になる。しかしながら、膨大となるファイルまたはファイル群とコントラクトとの紐付け管理やその検証方法が課題となる。

3. 提案方式

我々の提案方式をデータ構造とそのデータ構造を用いたファイル共有制御方法に分けて説明する。

3.1. データ構造

図 1 に示すように、我々の提案方式は、分散ファイルシステム上に、ファイルまたはファイル群とともに対応するコントラクトへのアクセス情報を保持している点が特徴である。制御対象のファイルと、コントラクトへのアクセス情報を同じオブジェクトに紐付けて管理している。このルートオブジェクトが、ブロックチェーン上のトークンと 1 対 1 で紐づく分散ファイルシステム上のオブジェクトとなる。1 つのトークンで管理するファイルの数だけ制御対象ファイル

Access control for IPFS based on tokens on blockchain
† Shigenori OHASHI, Hiroki WATANABE, Tatsuro ISHIDA, Shigeru FUJIMURA, Atsushi NAKADAIRA
† NTT Service Evolution Laboratories, NTT Corporation
‡ Junichi KISHIGAMI
‡ Muroran Institute of Technology

[2] https://storj.io/
[3] https://sia.tech/
[4] https://filecoin.io/

をルートオブジェクトに紐付けることで、ファイル群の取扱いも可能となる。当該トークンが含まれるコントラクトへのアクセス情報は、Ethereum[2]の場合、コントラクトアドレスやABIの情報になる。

ルートオブジェクトに、紐づくファイルのIDを含めることで、紐づけを行っている。ルートオブジェクトや制御対象ファイルなど分散ファイルシステム上のIDに、ハッシュ値を使用することで、受け取ったルートオブジェクトやファイルが要求したものであることを検証できる。これはIPFSなどの分散ファイルシステムの実装と同一である。

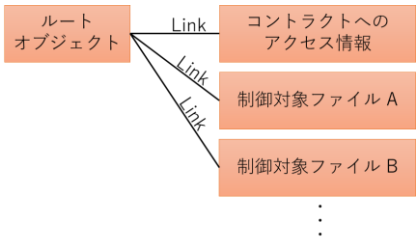

図 1: 分散ファイルシステム上のデータ構造

3.2. ファイルの共有制御方法

ファイルの共有は分散ファイルシステムを通じてファイルの要求を行い、要求を受信したノードが応じることでファイルが共有される。提案方式では、分散ファイルシステムのファイル要求メッセージ中にルートオブジェクトIDを含める工夫をしている。これにより要求メッセージを受信したノードはルートオブジェクトから紐づくコントラクトへのアクセス情報を特定できる。要求を受信したノードはコントラクトへのアクセス情報を使い、コントラクトを参照し、ファイルの共有前に要求者の資格検証を行うことができる。

4. 実装

フィージビリティ検証のため、IPFSをベースとして提案方式を実装した。評価用システムの全体像は図2に示す。

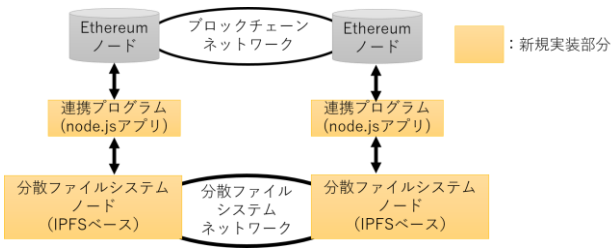

図 2: 評価用システムの概要

5. 結果

評価用システムにて、ファイルの追加、共有制御ができることを確認した。ブロックチェーンへのトランザクション処理を除き、連携プログラムおよび分散ファイルシステム側の処理に絞るとファイルの追加で平均392msec(既存のIPFSを使った同ファイルの追加で180msec)、ファイルの共有処理で406msec(既存のIPFSを使った同ファイルの取得で27msec)となった。トランザクション処理を含めるとファイルの追加処理では、コントラクトのコンパイル時間も含め、平均48885msec、共有制御処理全体では、平均4978msecとなり、ブロックチェーンへのトランザクション処理に掛かる時間が支配的となった。

6. 考察とまとめ

ファイル追加処理やファイル共有処理では、既存のIPFSに比べて時間を要しているが、ブロックチェーンと連動する分散ファイルシステムとして利用することを想定すると追加処理時間は相対的に小さく、許容範囲と考えられる。提案方式は分散ファイルシステムを起点としたブロックチェーンへのアクセスを実現できるため、不特定多数のユーザを想定したシステム、例えば、一般消費者も含めたユーザが作成したデジタルコンテンツの権利管理や流通管理のシステムなどへの適用が期待できる。

参考文献

[1] Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." (2008).
[2] Wood, Gavin. "Ethereum: A secure decentralised generalised transaction ledger." Ethereum project yellow paper 151 (2014): 1-32.
[3] Eberhardt, Jacob, and Stefan Tai. "On or off the blockchain? Insights on off-chaining computation and data." European Conference on Service-Oriented and Cloud Computing. Springer, Cham, 2017.
[4] Steichen, Mathis, et al. "Blockchain-Based, Decentralized Access Control for IPFS." The 2018 IEEE International Conference on Blockchain (Blockchain-2018). IEEE, 2018.
[5] Benet, Juan. "IPFS-content addressed, versioned, P2P file system." arXiv preprint arXiv:1407.3561 (2014).